In the Claims

Applicants have submitted a new claim set showing amended claims with insertions indicated by underlining and deletions indicated by strikeouts and/or double bracketing.

Please amend pending claims 1-6 and 8 as noted below.

Please add new claims 9-18 as shown below.

1. (Currently amended) A method of fast random access a mangement of a DRAM[[-type]] memory, including the steps of:

dividing the memory into memory banks accessible independently in read and write mode;

identifying an address [[@b] of ~~the~~ a bank concerned by a current request;

comparing the address of the bank concerned by [[a]] the current request with addresses of N-1 banks previously ~~required~~ requested, N being an integral number of cycles necessary to execute a request; and

if the address of the bank concerned by [[a]] the current request is equal to ~~the~~ an address of at least one of the N-1 banks previously requested ~~previous requests~~, suspending and storing the current request until the previous request involving the same bank is executed, otherwise, executing it.

2. (Currently amended) The method of claim 1, wherein the suspension operation includes stacking the requests in a ~~memory of~~ first-in/first-out ~~type~~ memory.

3. (Currently amended) The fast access DRAM management method of claim 1, further including for ~~the~~ data reading, the steps of:

storing in an output FIFO register ~~the~~ data read during the first M cycles of memory use; and

providing an output datum of the FIFO register, M cycles after each read request.

4. (Currently amended) The fast access DRAM management method of claim 1, wherein the memory is periodically refreshed line by line and bank by bank, and including the

step of comparing the address of the bank to be refreshed with the addresses of N-1 ongoing requests and of ~~the~~ N following requests and delaying ~~the refreshment~~ a refresh operation if the address of the bank to be refreshed corresponds to one of the bank addresses of the 2N-1 requests.

5. (Currently amended) The fast access DRAM management method of claim 4, including the steps of resuming the ~~refreshment~~ refresh operation and interrupting the request succession after a determined number of refresh cycle interruptions have occurred.

6. (Currently amended) ~~The fast access DRAM management method of claim 1, including the steps of~~[[:]]

A method of fast random access management of a DRAM memory, including the steps of:

dividing the memory into memory banks accessible independently in read and write mode;

identifying an address of ~~the~~ a bank concerned by a current request;

comparing the address of the bank concerned by the current request with addresses of N-1 banks previously requested, N being an integral number of cycles necessary to execute a request, and if the address of the bank concerned by the current request is equal to ~~the~~ an address of at least one of the N-1 banks previously requested ~~previous requests~~, suspending and storing the current request until the previous request involving the same bank is executed, otherwise, executing it;

storing N requests following the current request;

if ~~the~~ execution of the current request is suspended, executing one of the N following requests not in conflict with ~~the~~ a request being executed; and

if the executed request is a read request, arranging back the read information in the order of the executed read requests.

7. (Original) The fast access DRAM management method of claim 1, wherein the memory banks are distributed into sets accessible in parallel, whereby each set statistically only needs to process half of the requests.

8. (Currently amended) The fast access DRAM management method of claim 1, wherein the memory banks are distributed into several groups, the memory banks of a same group sharing ~~the~~ a same bus, and wherein two requests can be simultaneously transmitted to two distinct groups.

9. (New) A fast random access DRAM memory comprising:

a plurality of memory banks accessible independently in read and write mode;

means for comparing an address of a memory bank of the plurality of memory banks corresponding to a current request with addresses of memory banks of the plurality of memory banks corresponding to N-1 previous requests, N being an integral number of cycles necessary to execute a request.

10. (New) The fast random access DRAM memory of claim 9, further comprising:

means for identifying the address of the memory bank corresponding to the current request.

11. (New) The fast random access DRAM memory of claim 9, further comprising:

means for, if the address of the memory bank corresponding to the current request is equal to an address of a memory bank corresponding to at least one of the N-1 previous requests, suspending and storing the current request until the at least one of the N-1 previous requests is executed, otherwise, executing it.

12. (New) The fast random access DRAM memory of claim 11, further comprising:

means for storing N requests following the current request, and, if execution of the current request is suspended, executing a following request of the N following requests corresponding to a memory bank having an address not equal to an address of a memory bank corresponding to the N-1 previous requests.

13. (New) The fast random access DRAM memory of claim 12, further comprising:

means for, if the executed request is a read request, arranging back the read information in the order of the executed read requests.

14. (New) The fast random access DRAM memory of claim 9, further comprising:

means for storing data read during the first M cycles of memory use; and

means for providing an output datum, M cycles after each read request.

15. (New) The fast random access DRAM memory of claim 9, further comprising:

means for performing a refresh operation comprising refreshing the plurality of memory banks line by line and bank by bank; and

means for comparing an address of a memory bank to be refreshed with addresses of memory banks corresponding to N-1 ongoing requests and addresses of memory banks corresponding to N following requests, and delaying the refresh operation if the address of the memory bank to be refreshed equals an address of at least one of the memory banks corresponding to the N-1 ongoing requests or the N following requests.

16. (New) The fast random access DRAM memory of claim 15, further comprising:

means for resuming the refresh operation and interrupting a succession of requests for a memory bank after the refresh operation has been delayed a predetermined number of times.

17. (New) The fast random access DRAM memory of claim 9, wherein the memory banks of the plurality of memory banks are distributed into sets accessible in parallel, whereby each set statistically only needs to process half of the requests.

18. (New) The fast random access DRAM memory of claim 9, wherein the memory banks of the plurality of memory banks are distributed into several groups, the memory banks of a same group sharing a same bus, and wherein two requests can be simultaneously transmitted to two distinct groups.